

Table Tip

Create a flexible key field by using a counter as only one component

Counters are the perfect data type for key fields. Because Access generates sequential numbers, counters are guaranteed to be unique. Furthermore, if you work on a network and you share the database with other users, counters ensure

that two users don't ever accidentally use the same key value.

However, counters have their drawbacks, too. You often want a key field to store data that's more meaningful than ordinary sequential numbers. Also, you may have to change key values. As you know, Access doesn't let you edit counter fields.

In this article, we'll show you a compromise. We'll present a technique for creating a macro that builds key fields from the other field entries in the record. You can ensure the key will always be unique by including a counter field as one of the key's components. Because the key field isn't actually a counter, you can update the values. Furthermore, because you incorporate other field entries, the value is descriptive.

Figure A

Table: Members						
Member ID	Member Name	Address	City	State	ZIP	Counter Field
BROWN0001	Brown, David	9420 Bunsen Pkwy	Louisville	KY	40220	1
JOHN0007	Johnson, Fred	1213 3rd Street	Louisville	KY	40202	7
JONE0002	Russ Jones	324 10th Street	Louisville	KY	40202	2
JONE0004	Janice Jones	555 4th Street	Louisville	KY	40202	4
KRAM0006	Krammer, Sara	534 6th Street	Louisville	KY	40202	6
MARC0005	March, Frank	123 8th Street	Louisville	KY	40202	5
SMIT0003	Tanya Smith	278 11th Street	Louisville	KY	40202	3

You want your data-entry form to generate Member ID values from the member's name and a counter value.

IN THIS ISSUE

- Create a flexible key field by using a counter as only one component 1
- Displaying combo box data in separate text box controls 5
- Be careful not to use property names as field names 8
- Removing the login security from the PIM application 10
- Disabling the ability to run a macro from the Database window 12
- A simpler not-in query 13
- Creating customer copies and file copies of the same invoice 15
- 1993 *Inside Microsoft Access* Index 15


An example

Before we describe the technique, let's clarify these ideas with an example. Suppose you want the Members table key value to consist of the first two letters of the member's name and a four-digit counter value. Figure A shows some sample data from such a Members table. You want to be able to edit the Member ID key values, and you also want Access to generate the key values for you.

The technique

Generally speaking, you'll create a macro that generates a key value from the record's other field entries. You'll then assign the macro to the After Update event property of your data-entry form. When you leave the new record or issue the Save Record

Table A

Key	Field Name	Data Type	Properties
	Member ID	Text	Field Size = 8
	Member Name	Text	Field Size = 50
	Address	Text	Field Size = 30
	City	Text	Field Size = 30
	State	Text	Field Size = 2
	ZIP	Text	Field Size = 10
	Counter Field	Counter	

command from the File menu, Access will generate the record's counter value and then run your macro. The macro will construct the key value and assign the result to the new record's key field.

Unfortunately, After Update is *not* the perfect event for this technique. After Update will occur when you update an existing record as well as when you post a new record. Therefore, your macro will run every time you save a modified record. Since you want the macro to generate a key value only for new records, the macro must be able to distinguish new records from existing ones. Otherwise, you won't be able to update the key field. If you modify the key field's control on the form and then post your changes, your macro will reset the key field.

To prevent this from happening, you define a default value for the field and design the macro to generate a key value only when the field contains the default value. If the field doesn't contain the default value, the macro can assume the record already has a valid key field.

For example, you can assign the string *New* to the field's Default Value property. Then, use the Condition column in your macro to check whether the key field contains *New* before generating a value.

At first, you might think you could avoid all this hassle by assigning the macro to the form's On Insert property. After all, you trigger this event only when you add a new record. However, On Insert occurs when you type the first character of a new record. Remember, you want to build the key value from entries in the record's other fields. When On Insert occurs, you haven't yet entered any information.

Building a data-entry system for the Members table

Let's work through the example we introduced earlier. Table A shows the structure of the Members table. Figure A on page 1 contains some sample data. We'll first create a form for this table. We'll then develop a macro that generates new

Inside MICROSOFT ACCESS™

Inside Microsoft Access (ISSN 1067-8204) is published monthly by The Cobb Group.

Prices: Domestic \$59/yr. (\$7.00 each)
Outside US \$79/yr. (\$8.50 each)

Phone: Toll free (800) 223-8720
Local (502) 491-1900
Customer Relations Fax (502) 491-8050
Editorial Department Fax (502) 491-4200

Address: You may address tips, special requests, and other correspondence to
The Editor, *Inside Microsoft Access*
9420 Bunsen Parkway, Suite 300
Louisville, KY 40220

For subscriptions, fulfillment questions, and requests for bulk orders,
address your letters to
Customer Relations
9420 Bunsen Parkway, Suite 300
Louisville, KY 40220

Advertising: For information about advertising in Cobb Group journals, contact
Tracee Bell Troutt at (800) 223-8720, ext. 430.

Postmaster: Second class postage is pending in Louisville, KY. Send address changes to
Inside Microsoft Access
P.O. Box 35160
Louisville, KY 40232

Copyright:

Authorized Canada Post International Publications Mail (Canadian Distribution)
Sales Agreement #XXXXXX CANADA GST #123669673. Send returns to
Canadian Direct Mailing Sys. Ltd., 920 Mercer Street, Windsor, Ontario, N9A
7C2. Printed in the USA.

Copyright © 1994, The Cobb Group. All rights reserved. *Inside Microsoft Access* is an independently produced publication of The Cobb Group. The Cobb Group reserves the right, with respect to submissions, to revise, republish, and authorize its readers to use the tips submitted for both personal and commercial use.

The Cobb Group, its logo, and the Satisfaction Guaranteed statement and seal are registered trademarks of The Cobb Group. *Inside Microsoft Access* is a trademark of The Cobb Group. Paradox is a registered trademark of Borland International. dBASE III and dBASE III PLUS are registered trademarks of Ashton-Tate, a Borland International company. Microsoft, MS-DOS, and Access are registered trademarks of Microsoft Corporation. Microsoft Windows and Word for Windows are trademarks of Microsoft Corporation.

Staff:

Editor-in-Chief David Brown
Editing Meredith Little
Polly Blakemore
Elizabeth Welch
Production Artists Maureen Spencer
Marguerite Stith
Design Karl Feige
Managing Editor Elayne Noltemeyer
Circulation Manager Marjorie Glassman
Editorial Director Jeff Yocom
Publishers Mark Crane
Jon Pyles

Back Issues:

To order back issues, call Customer Relations at (800) 223-8720. Back issues cost \$7 each, \$8.50 outside the US. You can pay with MasterCard, VISA, Discover, or American Express, or we can bill you. Please identify the issue you want by the month and year it was published. Customer Relations can also provide you with an issue-by-issue listing of all articles that have appeared in *Inside Microsoft Access*.

Member ID values and assign the macro to the form's After update property.

Creating the form

To create the form, highlight the Members table in the Database window and click the New Form button (📄) on the tool bar. In the New Form dialog box, click the Form-Wizards button. Then, in the following dialog box, select the Single-Column option and click OK. When the Single-Column wizard's first dialog box appears, click the Fast Forward button (➡) to generate the default form. In the final dialog box, click the Design button. Figure B shows the form that the wizard creates.

Next, set the Default Value property for the key field's text box control. Start by selecting the Member ID text box and then clicking the Properties button (🔧) on the tool bar. Default Value is the thirteenth property in the property sheet. Move to Default Value and enter the word *New*.

We'll next create the macro that generates the Member ID values. But first, save the form by pulling down the File menu and clicking the Save As... selection. Enter *Members Form* in the Save As dialog box and click OK.

Creating the Generate ID macro

Return to the Database window by pressing [F11], click the Macro button to list the database's macros, and then click the New button. When the Macro window appears, click the Condition button (🔍) on the tool bar to open the Condition column.

In the Condition cell, enter

```
[Member ID] = "New"
```

In the Action cell, type *SetValue* or select *SetValue* from the selection list. The *SetValue* action has two arguments—Item and Expression. Item is the quantity that receives a new value; Expression is the value that the action assigns to Item. Assign *[Member ID]* to the Item argument and enter the expression

```
UCase$(Left$([Member Name],2)) &  
➡ Format([Counter Field],"0000")
```

for Expression.

When you've finished, pull down the File menu and click the Save As... option.

In the Save As dialog box, enter *Generate ID* and click OK. Then, close the window.

Now return to Members Form and assign the Generate ID macro to the After Update property by either typing *Generate ID* in the After Update combo box or choosing it from the dropdown list. Make sure you assign the macro to the *form's* After Update property. If, when you return to the form, the property sheet displays a control's properties, click the white rectangle at the intersection of the rulers. The property sheet will then list the form's properties. Once you locate the After Update property, you can click its dropdown arrow and choose *Generate ID* from the selection list.

A few enhancements

Generate ID is the macro that's most important to our technique. However, you'll probably want to create the two other macros we're about to show you in order to make this technique more reliable.

First, you may want a macro that validates the Member Name field before letting you leave the record. Remember, the Generate ID macro draws the ID value's first two characters from Member Name. If you don't enter a member name, the key field will consist of only the four-digit number. After opening a new Macro window by using the same steps we described previously, enter the conditions and actions listed in Table B. Then, save the macro under the name *Validate Member Name* and close the window.

The Validate Member Name macro checks whether the current record's Member Name text box contains a Null value. If

Figure B

You create this form for the Members table to implement our example.

Table B

The Validate Member Name macro

Condition	Action	Action Argument
[Member Name] Is Null	MsgBox	Message = You must enter a member name Beep = Yes Type = Warning!
...	CancelEvent	

Figure C

The value *New* appears in the new record's Member ID field.

Figure D

When you save the new record, the macro will generate a key value.

so, the macro first displays the Warning! message box and then issues the CancelEvent action to cancel the record update.

You should assign the Validate Member Name macro to the form's Before Update property. Return to the form and click the white rectangle to display the form's properties in the property sheet. Next, move to the property sheet and find the Before Update property. Then, click the drop-down arrow and select Validate Member Name from the list of macros.


You may also want to create a macro that prevents you from moving to the Member ID text box until the Generate ID macro assigns a key value. If you don't develop such a macro, you may accidentally delete the *New* entry, which tells Generate ID that the record still needs a key value.

Create a new Macro window and open the Condition column. In the Condition cell, enter

[Member ID] = "New"

Because *New* is the Member ID text box's default value, this condition will evaluate to True only for new records that don't yet have a key value. In the Action cell, enter *SendKeys*. The SendKeys action has two arguments—Keystrokes and Wait. Assign the string {TAB} to the Keystroke argument and leave the default value *No* in the Wait argument. Save the macro with the name *Next Control On New* and close the window.

Now return to the form again, click on the Member ID text box, and move to the property sheet. Click the On Enter property's dropdown arrow and select *Next Control On New* from the list of macros.

That's the last change you need to make to the form. Pull down the File menu and select Save. Then, click the Form View button () on the tool bar to test the macros you've put in place.

Using Members Form

As you scroll through existing records, the form lets you move to the Member ID text box and even change the value. However, when you move to the blank record at the end of the table, the Member ID text box shows the value *New*, as shown in Figure C. If you try to enter Member ID, the Next Control On New macro moves you to the Member Name field.

After you enter the member name and address information and then leave the record, Access runs the Before Update macro Validate Member Name. As long as you've entered a name, Access generates a value for the counter field and posts the record to the table. Next, Access runs the After Update macro Generate ID, which builds the key value from the Member Name entry and the counter and assigns the result to Member ID. In our example, if you enter *Snider, Frank* into the Member Name field and save the record, the Generate ID macro will generate the key value SN0008, as shown in Figure D. The SN component comes from the first two letters of Snider; the 0008 piece comes from the counter field.

Conclusion

In this article, we showed you a technique for generating a table's key value. The technique offers the advantage of using a counter as the key field—but without the restrictions of counters. Although we presented the technique for a specific example, you should be able to apply it to your tables. ❖

Subscribe to Inside Microsoft Access Resource Disk!

Do you wish that you could experiment with the forms, reports, tables, macros, modules, and queries we regularly feature in *Inside Microsoft Access* but don't have the time or patience to create them? If so, you may want to subscribe to Inside Microsoft Access Resource Disk. Once you subscribe, we'll send you a disk loaded with all the useful tips featured in that month's *Inside Microsoft Access*. (See the articles marked with the disk icon.)

A six-month subscription to Inside Microsoft Access Resource Disk costs \$29. A full one-year subscription is \$49. If you don't want to subscribe but would like the forms, reports, tables, macros, modules, and queries in a particular issue of *Inside Microsoft Access*, you can purchase a single disk for only \$9.95.

To subscribe or order a specific month's disk, just call Customer Relations at (800) 223-8720. Outside the US, please call (502) 491-1900.

ORDER TODAY!

Is This Your Copy?

If Not, Order Your Subscription Today!



Enter my subscription for one year of
Inside Microsoft Access for only \$49. That's just \$4.08
a copy instead of the usual \$6 an issue!

Please bill me later.

Name _____

Firm _____

Address _____

City _____ State _____ Zip _____

Phone (_____) _____ S 2066

Outside U.S.: \$8.50 ea., \$69/yr.

VGCK

Inside
MICROSOFT ACCESS

NEW ORDER!



NO POSTAGE
NECESSARY
IF MAILED IN
THE UNITED
STATES

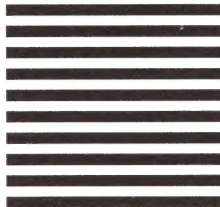
BUSINESS REPLY MAIL

FIRST CLASS MAIL

PERMIT NO. 618

LOUISVILLE, KENTUCKY

POSTAGE WILL BE PAID BY ADDRESSEE



THE COBB GROUP
PO BOX 35160
LOUISVILLE KY 40232-9719



Displaying combo box data in separate text box controls

When you create a combo box to look up values from another table or query, you can display as many of the table's or query's fields in the selection list as you want. However, the combo box control will display only one of those fields after you choose a value from the selection list.

As useful as combo boxes are, you may sometimes wish you could do more with them. Often, you might want to display on the form the other fields the combo box displays in the selection list.

Well, although combo boxes can display only one field, they retain their connections to the other fields. In this article, we'll show you how to create additional text box controls that display the hidden values in the selection list columns. The order form shown in Figure A demonstrates how using our technique can enhance your forms. When you select a customer in the Customer ID field's combo box, the text boxes in the panel on the right display additional customer information.

Comments on creating the combo box

Before we show you how to create the figure's order form, we must review a couple of important combo box properties. You use the Row Source property to define the particular table or query the combo box will display in its selection list. Also, you use the Column Count property to set the number of fields from the table or query you want to include in the combo box's selection list.

For example, suppose you create a combo box named CustomerLookup that lets you select a customer ID while viewing the first ten fields in the Customer table. To create this control, you'd set Row Source to *Customer* and Column Count to 10.

Displaying additional combo box fields

Now we'll describe how to use the combo box's Column property to display the other values in the selection list. You may not

know about this property, because it doesn't show up in the property sheet. You can access it only in expressions.

To access a particular column, you build an expression that begins with the combo box's control name, followed by a period and the *Column* keyword, and finally the column number enclosed in parentheses. Because the column numbers begin with 0, you use 0 to reference the first column, 1 to reference the second column, and so on.

To display the value in a text box, you assign your expression to the text box's Control Source property. Suppose you want to display the third column of the CustomerLookup combo box in a text box control. You'd first make sure the combo box's Column Count property is set to at least 3. Then, you'd create the text box on the form and type

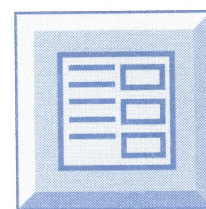
```
=CustomerLookup.Column(2)
```

into the text box's Control Source property. When you select a customer from the CustomerLookup combo box while viewing the form, the value in that row's third column will appear in your new text box.

Keep in mind that you're assigning a property value in this case—not a field entry. Therefore, you must place an equal sign in front of the expression.

An example

To illustrate how to implement this technique, let's build the order form shown in Figure A. The form displays basic order data from the Orders table and uses the



Form Tip

Figure A

The panel on the right of this form displays information for the customer whose name you select in the combo box.

combo box to look up data from the Customer table.

To make life easy, we'll use the Orders and Customers tables from the NWIND.MDB database that Microsoft ships with Access. Import the tables into your test database by using the File menu's Import... command.

Table A

Question	Response
Which field(s) do you want to include on your form?	Order ID Customer ID Order Date Required Date Shipped Date Order Amount
What kind of look do you want for your form?	Standard
What title do you want for your form?	Basic Order Info

In the menu command's first dialog box, select Microsoft Access as the file format from which you'll import objects. Then, select the NWIND.MDB database from the next dialog box. Finally, in the following dialog box's list of tables, select Orders and Customers.

Now that the tables reside in your test database, let's create the order form. Start by highlighting the Orders table in the Database window and clicking the New Form button (📄) in the tool bar. In the New Form dialog box, click the Form Wizards button and then select the Single-Column option in the following dialog box. In the series of dialog boxes the wizards present, make the selections and choices shown in Table A. In the last dialog box, click the Design button. The form wizard will generate our sample form shown in Figure B.

The form wizard will create default text boxes for all the fields you include. Before continuing, you'll need to replace the Customer ID text box with a combo box. First, remove the Customer ID text box by selecting the control and pressing the [Del] key. Then, open the field list and tool box. (You open the field list by clicking the Field List button (📋) on the tool bar. You open the tool box by checking the Toolbox option on the View menu.) Next, select the Combo box tool (📄) in the tool box, click Customer ID in the field list, and drag the mouse pointer to the former position of the deleted Customer ID text box. When you release the mouse button, a new combo box will appear, as shown in Figure C.

Next, configure the new combo box by using the property sheet. Start by clicking the Properties button (🔧) on the tool bar. First, make sure the property sheet is displaying the combo box's properties. Then set the properties listed in Table B. Your property sheet should resemble the one shown in Figure D.

You'll probably want to place the new combo box in the Customer ID field's position in the tab order. Pull down the Edit menu and select the Tab Order... command. Next, click the CustomerLookUp entry's row selector. Then, click again and drag the entry to just below the Order ID item's position. After you've restored the original tab order, click OK.

Figure B

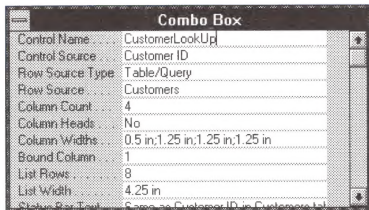
The Single-Column form wizard will generate this default form.

Figure C

Replace the Customer ID text box with a combo box.

Table B

Property	Value
Control Name	CustomerLookUp
Row Source	Customers
Column Count	4
Column Widths	0.5 in; 1.25 in; 1.25 in; 1.25 in
List Width	4.25 in

Figure D

Enter the combo box properties as illustrated here.

Finally, you must create the text boxes and corresponding labels that will display the data in the columns of the combo box's selection list. First, use the Text Box tool to add three unbound text box controls to the right of the Orders table's field controls. Then, change the text in the associated label controls to the three field names you'll display—*Customer Name*;, *Contact Name*;, and *Contact Title*:. You can change label text by clicking in the label until you get the insertion point cursor that lets you enter text. Arrange these controls as shown in Figure E.

Next, assign the CustomerLookUp combo box's Column properties to the text boxes' Control Source properties. Note that the Customer Name, Contact Name, and Contact Title fields reside in columns 1, 2, and 3, respectively. Assign the Column properties as shown in Table C. (Remember to begin the expressions with the equal signs.)

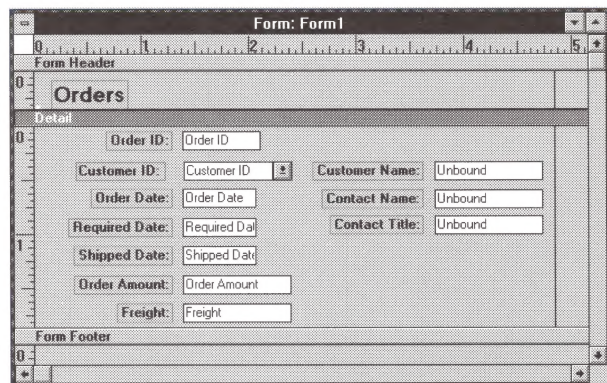
Now you're ready to view the form. First, save it with the File menu's Save As... command. In the Save As dialog box, enter *Orders Form* and click OK. Then, click the Form View button () on the tool bar. Access will display the first Orders record in the form. As we explained, the information in the combo box's selection list appears in the unbound text boxes.

To select another customer for the order, click the Customer ID combo box's drop-

down arrow. The selection list will appear as shown in Figure F. When you select a customer—say, *BERGS*—the new customer's data pops into the Customer Info panel.

Conclusion

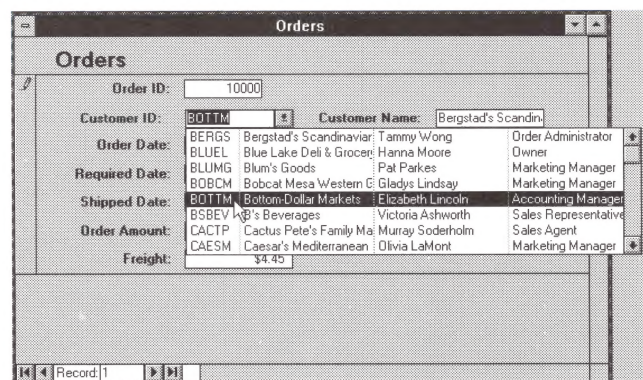
In this article, we showed you how to display in separate text boxes the data in a combo box's selection list. By using this technique, you can keep information from another table visible while you work with the data from the form's main table. ♦

Figure E

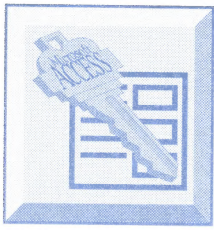
Arrange the label and text box controls to the right of the Orders table's field controls.

Table C

Text Box	Expression
Customer Name	=CustomerLookUp.Column(1)
Contact Name	=CustomerLookUp.Column(2)
Contact Title	=CustomerLookUp.Column(3)

Figure F

When you click the dropdown arrow, the selection list appears.



Be careful not to use property names as field names

Access lays down several rules you must follow in naming fields and other objects. For instance, you can use up to 64 characters as long as you don't include periods (.), exclamation points (!), or brackets ([]). Also, you can't begin a name with a space or a control character.

However, these rules don't tell the whole story: You can still get into trouble. In this article, we'll show you what can occur when you choose a field name that also happens to be a property name. In a nutshell, Access will let you use the name for the field, but it won't let you use the field

in expressions you may want to display on forms or reports. When you assign the expression to the text box control to display, Access will always mistake the field for the property that has the same name.

The Part Inventory table example

Now we'll show you how this problem can arise. Suppose you use a table named Part Inventory to store a list of parts and their dimensions and costs. Table A shows the table's structure; Figure A shows some sample data. Notice the name of the third field—WIDTH. A conflict arises when you use this name because form and report objects as well as the controls you place on forms and reports all have Width properties.

To create this table, list the tables in the Database window. Then, click the New button. In the Table window that appears, type the field names and data types as listed in Table A. Next, make the PART # field the table's key by first clicking the row selector and then clicking the Primary Key button (🔑) on the tool bar. Finally, use the File menu's Save As... command to save the table with the name *Part Inventory*, and close the window.

At first glance, the field names seem to work without causing any problems. As you can see from the figure's datasheet, Access stores data in the WIDTH column normally. However, trouble will arise when you create forms and reports and include the WIDTH field in an unbound control's expression. Access will use the property value rather than the contents of the field to compute the value of the expression.

Creating a Part Inventory form

To see this problem for yourself, create a form for the Part Inventory table. Highlight the table name in the Database window and click the New Form button (📄) on the tool bar. In the New Form dialog box, click the FormWizard button. Then, in the dialog box that follows, select Single-Column and click the OK button. The Single-Column

Table A

The Part Inventory table


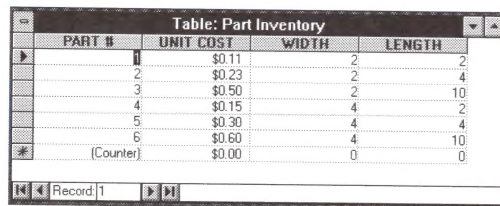
Key	Field Name	Data Type
	PART #	Counter
	UNIT COST	Currency
	WIDTH	Number
	LENGTH	Number

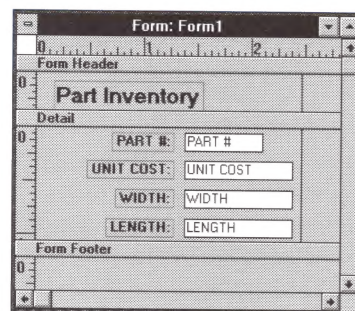
Figure A



PART #	UNIT COST	WIDTH	LENGTH
1	\$0.11	2	2
2	\$0.23	2	4
3	\$0.50	2	10
4	\$0.15	4	2
5	\$0.30	4	4
6	\$0.60	4	10
(Counter)	\$0.00	0	0

We'll use this sample data in our example.

Figure B



Form: Form1

Form Header

Part Inventory

Detail

PART #: PART #

UNIT COST: UNIT COST

WIDTH: WIDTH

LENGTH: LENGTH

Form Footer

We'll use this form to demonstrate the problems that result from naming a field WIDTH.

Microsoft Access™

Spotlight

Advertising Supplement

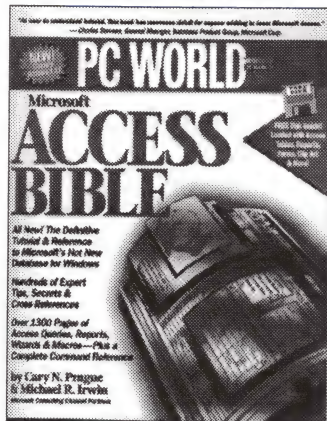
The Cobb Group
March 1994



**Unlock Good Fortune
With These Access
Products & Services!**

The Best Access Books Today!!

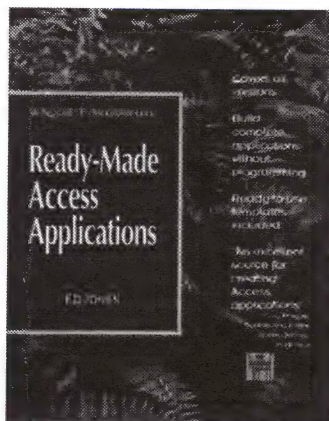
End/Power User



Access Bible
Prague/Irwin, IDG Books
1300 Pages, Includes Disk
\$34.95



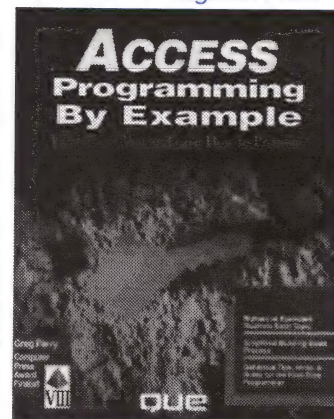
Power-User



Ready Made Access Applications
Ed Jones, McGraw Hill
415 Pages, Includes Disk
\$36.95

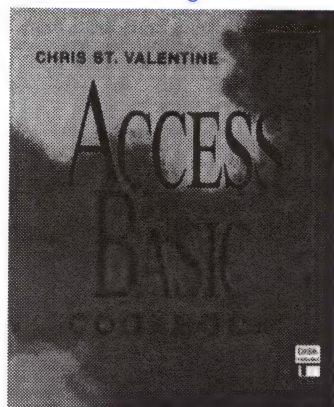


Novice Programmer



Access Programming by Example
Greg Perry, Que
600 Pages
\$27.95

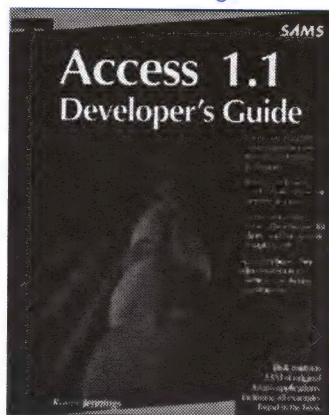
Novice Programmer



Access Basic Cookbook
St. Valentine, Addison-Wesley
400 Pages, Includes Disk
\$34.95

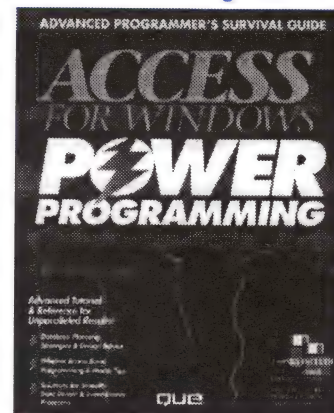


Advanced Programmer



Access Developers Guide
Roger Jennings, Sams
1000 Pages, Includes Disk
\$37.95

Advanced Programmer

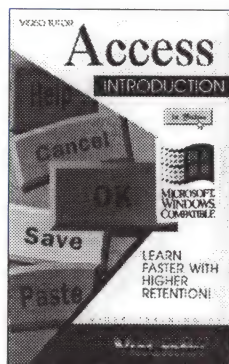


Access Power Programming
Liczbanski, Que
1000 Pages, Includes Disk
\$42.95

Special! The Video Bundle

Introduction
and Advanced
Both Videos...

**Only
\$89.95!**



Includes 2 Disks!!

US Phone Orders: (800) 277-3117

☐ **YES! I want to order.**

☐ **Please send your complete detailed catalog.**

Qty	Item	Price	Total
___	Access Bible	\$34.95	_____
___	Ready Made Access Apps.	\$36.95	_____
___	Programming by Example	\$27.95	_____
___	Access Basic Cookbook	\$34.95	_____
___	Access Developers Guide	\$37.95	_____
___	Access Power Programming	\$42.95	_____
___	The Access Video Bundle	\$89.95	_____
___	Bonus - 250 Access Hot Tips	\$12.50	_____
___	Bonus - Instant Reference	\$ 9.95	_____
Subtotal:			_____

US Shipping & Handling: **\$4.95***



* Canadian Orders: Add \$10.00 per book
* Europe/S. America: Add \$25.00 per book
* Pacific Rim/Africa: Add \$35.00 per book

**Fax and International Orders
Call (203) 644-5891**

Mail/Fax Orders: Fill in the Form Below and Send To:

Cary Prague Books and Software
60 Krawski Dr.
S. Windsor, CT 06074

Name _____
Company _____
Address _____
City _____ State _____ Zip _____
Country _____ Phone _____
Payment: ☐ Check ☐ Visa ☐ MasterCard ☐ Amex
Card# _____ Exp. Date _____
Signature _____

*Please Allow 5 days for Delivery
Overnight Shipping Available*

The Best Access Add-Ons!!



Mail Merge Report Wizard Version 2.0

Only **\$79.95**

- 🔑 Create Form Letters in Access in Eight Easy Wizard Screens
- 🔑 No Need to Use an External Word Processor
- 🔑 Creates a Standard Access Report
- 🔑 Works Just Like Other Access Report Wizards
- 🔑 Customize the Completed Mail-Merge Letter
- NEW Easily Change the Font Used For the Report Design
- NEW Save the Wizard Screen Entries for Later Reuse
- NEW Works With the Access Runtime Version
- 🔑 Control Date Placement and Format
- 🔑 Complete Documentation and Secrets Guide

Mail Merge Report Wizard Version 2.0

What do you want in each paragraph? Enter text and fields. Select a field format if desired for each field. Use the "Enter" key to add a blank line between paragraphs.

Available Fields: Date of Last Visit, Customer Number, Ms/Mrs/Mr, First Name, Last Name, Address, City

Field Format: January 31, 1993

Paragraph Appearance: We'd just like to remind you that it is once again time for a visit to the Dental Center! You are due for a visit every [Visit Interval], and your last visit was Format([Date of Last Visit], "mmmm dd, yyyy") for a [Services Rendered]. Please call us at your earliest convenience to schedule an appointment with [Service Provider].

If any of your insurance information has changed, please let us know when you call to make your appointment. Our records indicate that your dental insurance carrier is [Insurance Carrier], and the

Buttons: Open, Save, Cancel, Back, Next

Ready to Use Access Applications

Company Name
Company Slogan
Enter Address Here
City, State, Zip/Postal Code
Phone: (000) 000-0000

INVOICE

SOLD TO: Fred Smith
10 Jackson Lane
Mainville, NE 67708 USA
(555) 555-1388

SHIPPED TO: Sam Jones
25 Henry Ave
Montford, MI A-205 Canada
(555) 555-7744

INVOICE # 12345
INVOICE DATE 3/31/93
YOUR ORDER # 84456
TERMS Net 10
SALESMAN George
F.O.B. Hartford, CT

ITEM #	Quantity	Description	Price	Amount
12223	10	#2 Pine Boards	\$12.23	\$122.30
14206	5	Nails	\$5.01	\$25.05
14456	7	Screws	\$12.34	\$86.38
17885	1	Mail Gun	\$65.00	\$65.00
24553	7	Hammers	\$37.23	\$260.61
26453	3	Gadgets	\$10.00	\$30.00
41165	5	Halls	\$43.23	\$216.15
63342	15	Roof Truss	\$12.23	\$183.45
SUBTOTAL				\$7,272.48
TAX				\$42.00
FREIGHT				\$5.50
PAY THIS AMOUNT				\$7,319.98

MAKE CHECKS PAYABLE TO: [Name] [Phone Number] [Enter Company Name Here]

THANK YOU FOR YOUR BUSINESS!

The Access Business Forms Library

This is a library of professionally designed business forms that you can use in a standalone Access database container or integrate into your own Access applications. You can distribute individual forms from the library royalty free.

Each application includes all tables, forms, reports, and macros for instant use! Because each table, form, and report are standard Access objects, you can quickly and easily customize them to your own needs. Also included is a 120 page detailed user guide describing business uses of each form as well as a picture of each form and a customization guide to teach you how each form was created.

Only **\$79.95**



You Get All These!!!

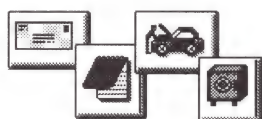
- About Business Forms
- Card and Gift Tracker
- Cash Receipts
- Check Requisition Form
- Commission Report
- Conference Room Scheduler
- Contributions Tracker
- Credit Application
- Customer Registration
- Expense Form
- Final Notice

- Back Order Letter
- Past Due Notice
- PC Tracker
- Address List
- Personnel Record
- Price List
- Prospecting Form
- Purchase Order
- Quotation
- Receiving Report
- Credit Reminder
- Resume Receipt
- Household Inventory
- Invoice
- Job Listing
- Job Application
- Bill of Lading
- Telephone Messages
- RFP/RFQ
- Sales Order
- Seminar Registration
- Shipping Order
- Videotape Log
- Wedding Planner

Also available...

The Button Library

1200 Command Buttons for Customizing Access Forms!



NEW! Special Price, \$99.95

See our other ad in Inside Access for special offers on our books and training videos!

US Phone Orders: (800) 277-3117

Fax and International Orders
Call (203) 644-5891

- ☐ Yes! I want to order.
- ☐ No. I don't wish to order at this time. Please send me detailed information on these products at no charge.

Qty	Item	Price	Total
___	Access Business Forms Library	\$79.95	___
___	Mail Merge Wizard Deluxe	\$79.95	___
___	Mail Merge Wizard Update	\$29.95	___
___	The Button Bundle	\$99.95	___
___	Forms and Buttons Sampler	\$ 9.95	___
		Subtotal:	___
		Shipping & Handling:	\$4.95*
		Total:	___



* Canadian Orders: Add \$10.00
* Europe/S. America: Add \$25.00
* Pacific Rim/Africa: Add \$35.00

Mail/Fax Orders: Fill in the Form Below and Send To:
Cary Prague Books and Software

60 Krawski Dr.
S. Windsor, CT 06074

Name _____
Company _____
Address _____
City _____ State _____ Zip _____
Country _____ Phone _____
Payment: ☐ Check ☐ Visa ☐ MasterCard ☐ Amex
Card# _____ Exp. Date _____
Signature _____

All Software Shipped 2nd Day
Overnight Shipping Available

WIND

Windows Information Network for Developers

Join the only BBS dedicated to supporting developers of Access, Visual Basic, C++ and SQL.

WIND offers: a full messaging system for questions, tips and tricks; file areas filled with freeware, shareware and demo programs; and, developers' books and commercial software products.

To take a free look at WIND, dial-in to (216) 694-5734.

WIND
200 Public Square, Ste. 26-4600
Cleveland, Ohio 44114

Advertiser's Index

Company	Page #
Cary Prague Books & Software	AS-2,3
CompuLAN.....	AS-6
Crystal Software.....	AS-6
FMS, Inc.....	AS-5
ProData	AS-6
SPECTRUM Human Resource Sys.	AS-4
TEAMWorks Technologies.....	AS-8
Tippecanoe Systems.....	AS-6
Windows Info. Network for Developers	AS-4
Winnovation Group	AS-4
ZD Press	AS-7

Microsoft Access SPOTLIGHT is published by The Cobb Group. For advertising rates and information, contact Tracee Bell Troutt (ext. 430) at (800) 223-8720. Copyright © 1994 The Cobb Group.

MASTER ACCESS!

PAUL NIELSEN COMPUTING provides you with the tools and the understanding to master Access.

Team Access - A complete Project Manager for Access
- Object Level Version Control
- Developer Progress Documentation
- Multi-Project Multi-Developer On-Line Environment

PowerShell - Powerful Add-in Library - turns plain applications into bulletproof and robust applications

On-Site Training Courses

Enhance your working knowledge of Access with any of the on-site training and courseware offered by PAUL NIELSEN COMPUTING.

An industry leader in the Access community, Paul Nielsen is the Technical Editor of Access Advisor Magazine, and has addressed national Access conventions and seminars.

TeamAccess
5 User License
\$295 Version 2.0
PowerShell - \$95
Version 2.0

PAUL
NIELSEN
COMPUTING

6011 Willowbottom Drive Hickory, NC 28602

1-800-648-2913
International
1-706-865-5115



HRVantage™

The Power of Windows™ Technology for HR Professionals

From recruitment management to applicant tracking, from benefits administration to government compliance, the tools you need for organizational control are just a mouse click away.

Built on Microsoft's® Access® database platform, HRVantage® provides full-featured functionality via easy data entry and retrieval, advanced analysis, unlimited history tracking and powerful graphical reporting. For a detailed look into the power and flexibility of HRVantage, call Spectrum today at:

1-800-4SPECTRUM

SPECTRUM

Human Resource Systems Corporation
1625 Broadway • Suite 2700 • Denver, CO 80202



Database
Documentation
Program

for
Microsoft Access®



Comprehensive Documentation and Analysis of Microsoft Access® Databases!

Microsoft Access offers a sophisticated new way to develop Windows applications. You can easily build complex forms, reports, queries and macros. However, with this power comes complexity. As you add features to your applications, understanding them becomes more difficult. Total Access provides the documentation you need to take control of your databases. From simple features such as printing macros and table structures to more complex documentation such as macro calls and fields used across objects, Total Access allows you to create Microsoft Access applications more efficiently.

Documents:

- Tables
- Fields
- Queries
- Forms
- Reports
- Controls
- Field Cross-Reference
- Table Relationships
- Macros and Calls
- Object Inventory
- Descriptions
- Properties *and more!*

Total Access provides the information you need to develop and maintain Access applications more productively.

Features:

- Entirely menu driven.
- Runs from within Microsoft Access
- Documents all objects in one pass.
- All documentation placed in Access tables.
- View data using our sophisticated forms.
- Includes over 40 built-in reports.

Total Access places all output in Access tables so you can use Access to manage the documentation. View, sort or query the results. Use our reports or create your own!

"Total Access is an indispensable utility that comprehensively documents Access databases....Every serious Access user should own a copy.... Total Access may just be the Access add-on of the year."

Paul Litwin, Editor, *Smart Access*

Total Access is available now at the introductory price of \$145 + \$5 S&H (\$15 Int'l). That's \$50 off the regular price of \$195. Virginia residents add 4.5% sales tax.

30 day money back guarantee!



FMS, Inc.

8027 Leesburg Pike
Suite 410, Dept. 322
Vienna, VA 22182, USA

(703) 356-4700 ext. 322
FAX (703) 448-3861, Compuserve 73710,463

If You Type 2000 WPM...

You Don't Need a Plustek
Full-Page Color Scanner!

Only **\$449.95**—An Incredible Deal!

Includes WordScan OCR Picture Publisher LE

- 1200 DPI
- HP Emulation
- 16.7 Million Colors
- Twain Compliant
- 24-bit True Color
- 2-year Warranty

CompuLAN Center

1-800-379-7226 ■ 1-214-245-5130

Visa and MasterCard orders welcome

Visual DDF for Btrieve

Do you have a Btrieve data file and don't know the layout? Do you want to create the data dictionary (DDF) files and attach to Access or VB 3.0? Visual DDF from Prodata can help.

Key	Seq	Position	Length	Key Type	Key Flags
0	1	1	6	String	307
1	2	7	8	String	308
2	1	1	6	String	309
3	2	27	10	String	310
4	2	1	23	Integer	256

File Specifications

Page Size
Record Length
Number of Indexes
Number of Records
File Flags
Fixed & Variable

Index Specifications

Multi-segment
Position & Length
Data Type
Key Flags

Any valid Btrieve data file can be opened. The file specifications and a complete layout of the indexes are displayed. Individual records in the file can be retrieved and analyzed to determine the type of data on a byte by byte basis. After the record layout has been mapped, you can create the standard *File*, *Field*, and *Index* DDF files. With these files, you can attach to Microsoft Access. *Special Spotlight price—\$99.00.*

Prodata

12101 Menaul Blvd NE • Albuquerque, NM 87112
(505) 294-1530 • Fax (505) 298-3757

Access Communications

Now you can write an Access Module that allows you more *access* with Access. Do you need to make your Access Application communicate over a **COM port**? Need your Access Data Base to be a client or server or transfer data files? Need to have your application talk to a modem, a mainframe, another PC, or custom hardware? Now you can do this and more using **CrystalCOMM**.


CrystalCOMM for Windows \$175

CyrytalCOMM for NT \$200

CrystalCOMM is available on the Windows and NT environments. It supports development of modem or serial port communication programs. It includes everything to maintain consistency in the Windows foreground or background environment and includes DLL support. CrystalCOMM supports **XMODEM**, **XMODEM-CRC**, **XMODEM-1K**, **YMODEM**, **YMODEM/Batch**, **ZMODEM**, **KERMIT**, and **ASCII** protocols through a simple, structured function interface. Supports **DigiBoard** and up to 9 Simultaneous ports at high speed. Detailed examples in Access, Visual Basic, C, and others. Access example includes client-server queries over communications links. Library includes object, optional source code and comprehensive documentation.

CrystalDial for Windows \$45

Only want to dial phone numbers from your Access database form? Call **CrystalDial** to dial from your application.

 **Crystal Software, Inc.** 906 822-7994

P. O. Box 43, Amasa, MI 49903, USA - FAX (906) 822-7994

BARS & STRIPES



Just **\$99⁰⁰**

**New! . . . Bar Codes for
Microsoft® Access™**

- Unique bar coding software application that installs directly inside Microsoft's Access
- Also available, versions for Microsoft Word™ for Windows and WordPerfect™ for Windows
- Choose from Code 39, UPC-A, UPC-E, Postal, Code 93, Code 128, Codabar, EAN-8, EAN-13, or Interleaved 2 of 5
- DLL Toolkit Available
- Not a font. Bar code images are created. No clipboard compatibility hassles.



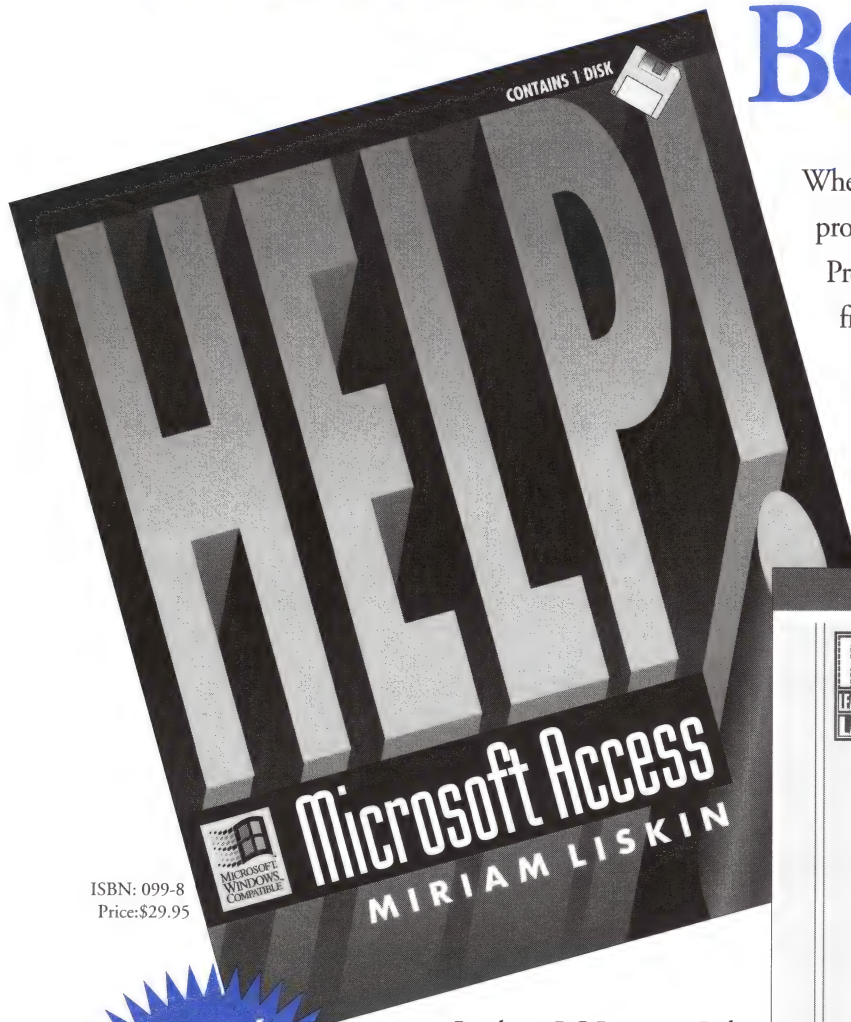
Bar Code
Decoder &
Wand -- \$199

 **TIPPECANOE**

 **510.416.8510**
Visa and Mastercard Accepted



SUPER SAVINGS ON MICROSOFT ACCESS BOOKS.



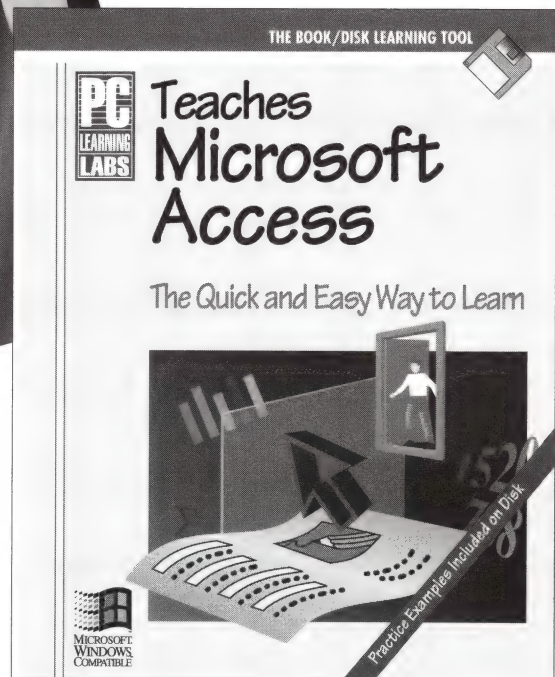
ISBN: 099-8
Price: \$29.95



Look to *PC Learning Labs* *Teaches Microsoft Access* for the tried and tested learning techniques that are guaranteed to bring new users up to speed on the newest version of Access.

Whether you're a first-time user or a seasoned professional, the authorities at Ziff-Davis Press can guide you to getting the most from Microsoft Access.

Turn to best-selling author Miriam Liskin. *HELP! Microsoft Access* tackles the application with a comprehensive, yet easy-to-use format.



ISBN: 225-7 Price: \$22.95 Available April

All Ziff-Davis Press books come with a 30-Day Money-Back No-Risk Guarantee.
Corporate and Academic discounts are available.

To get The Cobb Group's special 20% Discount
plus FREE SHIPPING call 1-800-688-0448, ext. 176.

*Quantity discounts available. Sales tax will be added where applicable. Offer good only through this toll-free number.



© 1994 Ziff-Davis Press

Wouldn't it be great to "copy" an image of a document right into a Microsoft Access database and then file it, annotate it, fax it, print it, and *share it* - using a software package that costs only \$245?

\$145

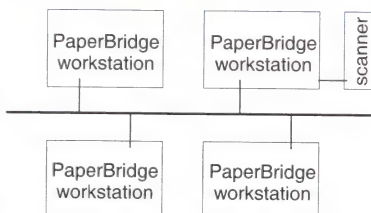
IMPOSSIBLE?

Document Image Management - On a PC! Now your desktop PC or workstation can perform document imaging tasks that previously required a much larger computer. Using new software technology you can copy documents into computer images and use them in place of paper. Routine office tasks can now be automated easily. Find documents faster and share them with other PC users. Make notes on them or store, fax, or e-mail them - all with PC document imaging management.

Microsoft
ACCESS

PaperBridge for the Access Database - On the PC. It is simply the best value in document imaging available anywhere. PaperBridge adds full-function document image management capabilities to Microsoft's Access relational database.

The result is industrial strength imaging seamlessly coupled with one of the most popular Windows database and application development environments. And PaperBridge offers other advanced features such as multi-page document display, user configurability, fast compression /decompression, annotation, and fax-in/fax-out. Altogether, a potent mix of features!



Powerful Peripheral and Network Support.

PaperBridge is office-ready. It supports most network and Windows-based client/server architectures. It also supports a full range of

imaging peripherals and subsystems including scanners, faxes, printers, print accelerators, and optical storage systems.

PAPERBRIDGE



September 1993

Easily Add Imaging to Access Applications.

Between PaperBridge and Access, development of complete Windows-based imaging applications couldn't be easier - or less expensive.

PaperBridge includes tools such as "wizards" and pre-tested macros to assist the novice developer as well as documentation and Access Basic routines for the experienced programmer. Access provides the rest. PaperBridge is the only open imaging development environment available. It's fast, complete and easy.

Ask for Your \$145 Introduction to Our \$245 Breakthrough.

Mention offer IA394 and you can try the developer version of PaperBridge for only \$145. The \$100 savings is a great start to PC-based imaging with Access. One to a customer. Site and run-time license pricing available.

Check it out!

NOT IMPOSSIBLE WITH PAPERBRIDGE. TRY IT.

Make me a believer! Send me PaperBridge for Access - NOW.

To get your copy of PaperBridge for Access simply mail this coupon with your check for \$145 plus \$6.50 shipping and handling to:



TEAMWORKS Technologies
65 Boston Post Rd. W., Marlboro, MA 01752

Or, for even faster service call toll-free:

1-800-532-3198

NAME		TITLE	
COMPANY			
STREET ADDRESS			
CITY, STATE, ZIP,			
PHONE NUMBER		FAX NUMBER	
VISA/MC <input type="checkbox"/>	AMEX <input type="checkbox"/>	DISCOVER <input type="checkbox"/>	Credit Card Number
SIGNATURE			Expiration

FormWizard's first dialog box will appear. Click the Fast Forward button (🚀) to tell the wizard to generate the default form. In the next and final dialog box, click the Design button. The FormWizard will generate the form shown in Figure B.

Now suppose you want to multiply the WIDTH and LENGTH fields and display the result on the form. To do so, you create a new text box control that displays the results of the calculation $[WIDTH] * [LENGTH]$.

First, lengthen the form's Detail section to make room for the new text box. Then, open the tool box if it isn't already on the screen. (You do so by checking the Toolbox option on the View menu.) Next, use the Text Box tool (📄) to place the new control just below the LENGTH text box, as shown in Figure C.

Now you want to assign the expression to the text box's Control Source property. Open the property sheet by clicking the Properties button (🔧) on the tool bar. Make sure the property sheet lists the new control's properties. Type the expression $= [WIDTH] * [LENGTH]$ into Control Source.

Before you press [Enter] to finalize the Control Source setting, your entry will look like the one in Figure D. However, once you press [Enter], Access replaces the [WIDTH] field name with the Width property name, as shown in Figure E.

When the expression multiplies the LENGTH field by the Width property, the text box's value will be wildly incorrect. Access stores the Width property value in twips—the base unit of measure for Windows' screen size. Figure F shows the non-sense result in our example. In this case, the text box labeled Field15 displays the result of multiplying the LENGTH field entry, 2, by the Width property value, 3560.

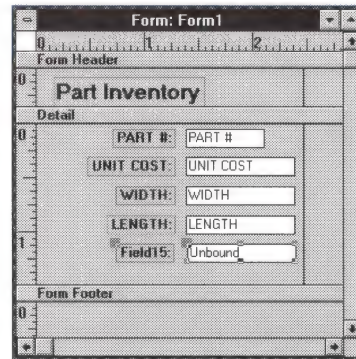
Suggestions

In summary, Access lets you use the name of a property as a field name. However, if you use the field name in an expression, Access will use the property value instead of the field value. The only way to avoid this problem is to be sure you don't use property names for your field names.

Unfortunately, a property name is sometimes the most obvious choice for a field name. If you find yourself in this situation, you should probably make some small

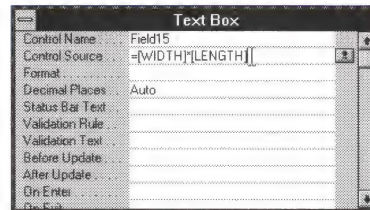
change to the field name to distinguish it from the property name. For example, you could end the field name with the pound sign (#). In this case, you could use WIDTH# instead of WIDTH. ❖

Figure C



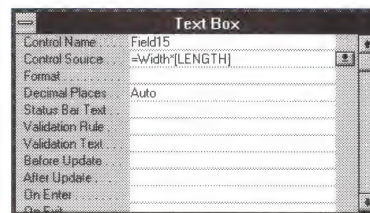
Place the new text box just below the form's LENGTH text box.

Figure D



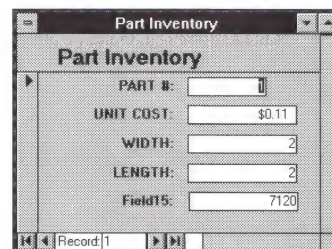
You can type the [WIDTH] field into the Control Source property's expression...

Figure E



...but Access will convert the field identifier [WIDTH] to the property name Width.

Figure F



The calculated value, labeled Field15, is meaningless when Access multiplies the LENGTH field entry by the Width property.



Removing the login security from the PIM application

If you've examined the sample databases Microsoft ships with Access, you've probably discovered the Personal Information Manager (PIM) application. Many users have found this application very useful for making things-to-do lists and keeping track of appointments. Furthermore, if you work on a network, the one database file can manage all users' appointments. That way, you can review your associates' appointments as you schedule a meeting.

When you open the PIM.MDB database file (located in the Access installation directory), several windows appear and disappear as the application loads. Then, the dialog box shown in Figure A appears and asks you to enter your user name and password. If you haven't used the PIM application, you'll need to use the sample user's name and password—John Doe and John, respectively. After you do so, the applica-

tion displays the main form shown in Figure B. You can then enter or review your appointments and so on.

As useful as this application is, the login process can be annoying if you're the only user and, as a result, have no need for the login security. In this article, we'll show you how to remove the login process so that the main form appears as soon as you launch the application.

Modifying the PIM application

The article "Disabling Individual Macro Statements," in the July 1993 issue of *Inside Microsoft Access*, proposed that you remove the login dialog box by eliminating a macro action that calls a function named DoLogin(). However, simply skipping that function has unfortunate side effects. For example, the function determines the current date, without which the application can't properly initialize the calendar display and appointment book.

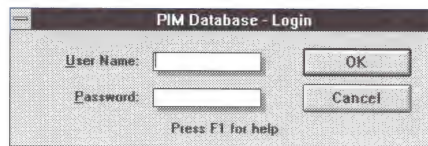
As you'll see, removing the login security doesn't require any major changes to the application. You just strip out the statements in DoLogin() that produce the PIM Database - Login dialog box and validate the user name and password. Instead of displaying the dialog box, your version of the DoLogin() function will directly insert your user name.

Start by opening the PIM.MDB database file while holding down the [Shift] key. You first pull down the File menu and click the Open Database... option. You next highlight the PIM.MDB file in the selection list, press and hold the [Shift] key, and click OK.

Entering your user information in the Person table

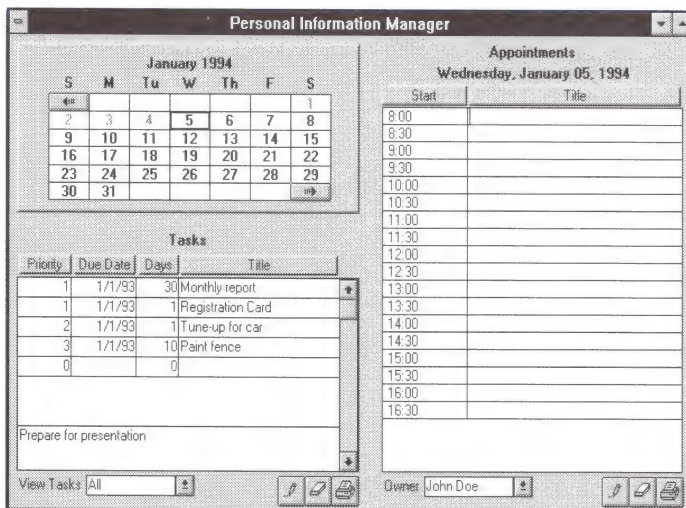
Before updating the DoLogin() function, you must insert your name in the application's Person table, which holds the list of users. While the Database window shows the list of tables, double-click the Person table. Then, move to the blank row at the end of the datasheet and enter your name in the Name field. Next, move to the Password field and type a password. (You won't ever

Figure A



When you launch the PIM application, you must log into the system by using this dialog box.

Figure B



After you log into the PIM application, this form appears.

use this password: Remember, you're removing login security. However, as a rule, you should provide passwords when asked for them.)

After you type the password, save your new record by selecting the Save Record command on the File menu. The counter field, ID, will generate your user ID value. Write down this number—you'll need it as you modify the DoLogin() function—and then close the Table window.

Updating the DoLogin() function

Next, return to the Database window and click the Module button to list the database's modules. Then, right-double-click the Login module to open the module in Design view. The Module window will appear, showing the declarations section. Display the DoLogin() function by clicking the Procedure combo box's dropdown arrow and choosing the function in the selection list. (You'll find this combo box in the tool bar.)

You'll modify the function mostly by omitting statements. However, we suggest that rather than just deleting the statements, you convert them to comments. To do so, insert an apostrophe at the beginning of the line. Listing A shows how your DoLogin() function should appear after you finish. Notice we

Listing A: DoLogin()

```
'-----
' FUNCTION: DoLogin
' PURPOSE: Provide verification of user login
'-----
Function DoLogin () As Integer
Dim db As Database, t As Table

    ' Open the login form as a modal dialog

'Cobb DoCmd OpenForm "Login", A_NORMAL, "", "", A_EDIT, A_DIALOG
'Cobb
'Cobb ' See if the user just closed the dialog
'Cobb
'Cobb If (Not (FIsLoaded("Login"))) Then
'Cobb DoError ideLOGINABORTED
'Cobb DoCmd SelectObject A_FORM, "Login", True
'Cobb If FToolBarWasUp() Then
'Cobb x% = ShowToolBar()
'Cobb End If
'Cobb DoCmd Close
'Cobb End If

'Cobb ' Validate the user and open the main form if valid.
'Cobb
'Cobb If (FValidUser(Forms!Login!UserID, Forms!Login!Password)) Then
'Cobb If (FIsLoaded("Login")) Then
'Cobb
        ' Update login information
        .....
'Cobb Replace ID = DLookup("ID", "Person", "[Name]=""" + Forms!Login!UserID + """)
        ID = 13
        .....

Set db = CurrentDB()
Set t = db.OpenTable("Current")
t.MoveFirst
t.Edit
t.Person = ID
t.Logon = ID
t.Date = Date
t.Update
t.Close
db.Close

'Cobb DoCmd Close A_FORM, "Login"
'Cobb End If
'Cobb DoCmd OpenForm "Daily", A_NORMAL, "", "", A_EDIT, A_NORMAL
'Cobb Else
'Cobb If (FIsLoaded("Login")) Then
'Cobb DoCmd Close A_FORM, "Login"
'Cobb End If
'Cobb f% = DoLogin()
'Cobb End If

End Function
```


inserted the word *Cobb* in addition to the apostrophe to distinguish the comments we're suggesting from the original comments.

We've highlighted the only line you must actually change. Replace the statement

```
ID = DLookup("ID", "Person", "[Name]=""" +  
Forms!Login!UserID + """)
```

with

```
ID = <ID value>
```

where *<ID value>* is the number that appeared in the ID field when you entered your name in the Person table. The number 13 appeared when we worked through the example. After you've made these changes, save the module with the File menu's Save command. Next, close the Module window and then the Database window.

You can now use the PIM application without having to log in. When you open the application, the main form appears without your having to use the PIM Database - Login dialog box. ♦



Macro Tip

Disabling the ability to run a macro from the Database window

You can create two types of macros in Microsoft Access. One type of macro automates tasks you perform from the Database window. For instance, you may create a macro that imports a text file into a table, specifying the type of file format and the destination table. Another type of macro you create—a macro group—is actually a collection of distinct macros. You usually develop these macros to collect the macros you use on a form or report.

You run the first type of macro by double-clicking the macro name in the Database window. However, you don't use a macro group that way. Access calls the macros that reside in a macro group in response to events that occur on the form or report. In fact, running a macro group from the Database window is annoying. Since you designed the macro actions in the group for a very specific set of circumstances, an error will almost always occur.

It'd be nice if you could avoid the errors in this situation. If you accidentally run a macro group from the Database window, you want to simply return to the Database window. In this article, we'll show you a simple tip for designing macro groups that prevents Access from executing the macro's actions when you run the macro group from the Database window. We'll start by reviewing macro groups and how you define several macros within a single macro object.

Understanding macro groups

As we mentioned, macro groups contain several individual macros that each handle a specific event on a form or report. You define several macros within a macro group by clustering each macro's actions and then using the Macro Name column to assign a name to each group of actions. To assign the macro names, you open the Macro Name column by clicking the Macro Names button (📁) on the tool bar and then enter the names next to the first action of each macro in the cluster.

The trick

What actually happens when you run a macro group from the Database window? Well, Access *doesn't* execute all the actions in the macro group. Instead, it executes only the actions in the group's first macro. When it encounters the second macro's name, it will stop.

You can use this fact to avoid the error messages that appear if you inadvertently run a macro group from the Database window. You simply begin the first macro on the second row rather than on the first. Access will consider the blank row as the macro group's first macro even though the row doesn't have a name. As a result, Access will stop running actions when it encounters your first macro's name in the second row.

An example

As an example, suppose you assembled into a macro group the three macros you created in “Creating a Flexible Key Field by Using a Counter as Only One Component” on page 1. This macro group would define the macros that control how you update the key field when certain events occur during data entry. You don’t ever run these macros from the Database window.

If you follow the advice we gave you in this article, you’ll leave the first row blank in these macros, as shown in Figure A. That way, Access will stop running the macro when it encounters the first macro name.

Providing online comments for macro groups

This basic tip for disabling a macro’s operation from the Database window leaves the door open for another useful technique: You can use the first row of a macro group to display a message box that describes the purpose of the macro group.

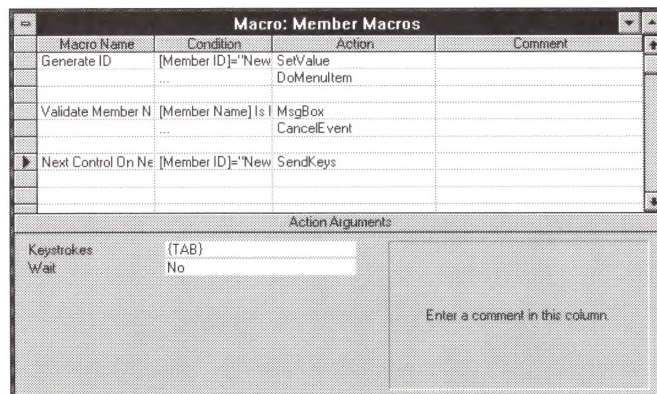
Returning to our example, Figure B shows the sample macro group after we inserted a MsgBox action that displays the message *This is a macro group for Members Form*. Figure B also shows the message box that appears on the screen.

Conclusion

In this article, we showed you how to avoid the annoying error messages that usually occur when you run a

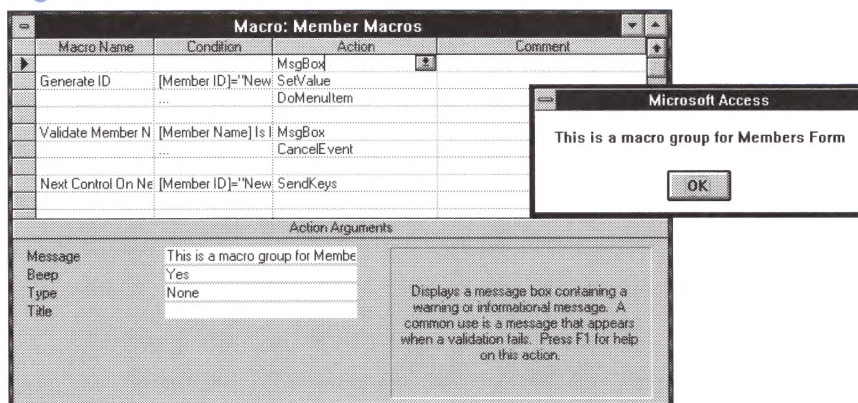
macro group from the Database window—you simply leave the macro grid’s first row blank. We also showed how you can use the first row to provide online comments for the form. ♦

Figure A



Leave the first line of the Customer Table Form macro blank to avoid errors in case you accidentally run the macro from the Database window.

Figure B



By using a MsgBox action in the macro group’s first row, you can display a description of the macro group.

A simpler not-in query

The article “Selecting Records from One Table That Aren’t in Another Table,” in the July 1993 issue, caught my eye recently. The article describes a not-in query that returns the records from one table that don’t have a matching record in another table. I have addressed this problem on my own, and I believe my solution is less complicated and produces a faster query.

First, I’ll review your method so that we can compare the two techniques. As the article describes, you use an outer join to

make sure the query can select all records from the larger table. You then group on the key field from the larger table and use the Count operation to count the records from the smaller table that have a key value matching those in the large table. The query will count either 1 (there is a matching key) or 0 (there is no matching key). You then select records that have a count of 0 in the smaller table.

With my technique, you also use an outer join to link the tables. However, you

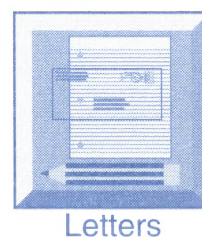
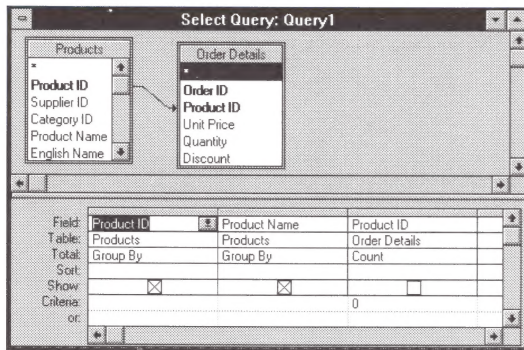
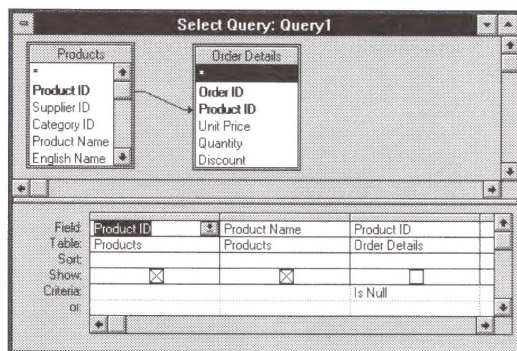


Figure A



The July article's method for creating a not-in query uses the Total row to select records in Order Details that have a count of 0.

Figure B



Instead of using the Total row, this method allows you to select records from Products that have a null entry in Order Details.

don't use the Count operation. Instead, you select records from the larger table only if the value of the smaller table's key field is Null. A Null key field means the table doesn't contain a record with a matching key value.

An example

As an example, let's create not-in queries in the NWIND.MDB sample database so that all readers can give it a try. I'll create a not-in query listing products that have never been ordered. This is a not-in query because you are looking for Product ID entries in the Products table that don't appear in the Order Details table.

The query shown in Figure A implements the original technique. To create this query, you complete the following steps:

1. Include the Products and Order Details tables in a new select query.
2. Convert the equi-join, which automatically appears, to an outer join so that the query can select all rec-

ords from the Products table. (Do so by double-clicking the join line and, in the Join Properties dialog box, selecting the option that includes all records from Products.)


3. Add fields from the Products table you want to see in the query's datasheet—Product ID and Product Name, for example.
4. Add the Product ID field from the Order Details table and deselect the column's Show box.
5. Click the Totals button () in the tool bar to open the Total row. In the Order Details table's Product ID column, replace the default value Group By with Count.
6. Enter 0 in the Criteria cell of the Order Details table's Product ID field.

Figure B shows my version of this query. To create it, you complete the first four steps I list above. But then, instead of opening the Total row, just enter *Is Null* in the Criteria cell of the Order Details table's Product ID field.

When you run this query, Access will return an empty datasheet. Nothing is wrong with the query: The Order Details table simply stores an order for every product. If you want the query to select records, you'll need to add extra products into the Products table.

Notes

As you can see, using the Is Null condition in the Criteria cell is much simpler and makes a faster query. Furthermore, the resulting record set is updatable. You can update product information directly from the query's datasheet.

*Oleg Ostrozhansky
Evergreen Park, Illinois*

We thank Mr. Ostrozhansky for his much improved not-in query. If you've used our version of the not-in query, you may have been frustrated by it. One problem is that you can't base other queries on queries that are not updatable. Now that you can create an updatable not-in query, you can incorporate the not-in query in more sophisticated queries.

Creating customer copies and file copies of the same invoice

I would like to know if there is an easy way to print duplicate copies of a report while changing a single label for each copy. Specifically, I want to create a customer copy and file copy of each invoice I print. Other than creating two reports and printing one after the other, I can't think of a straightforward way to do it.

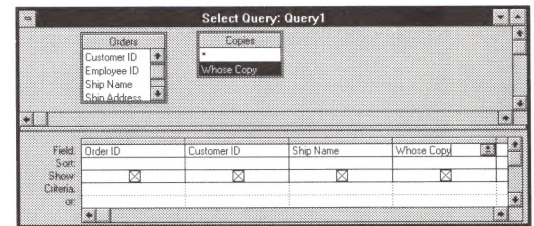
Jay Ritchie
Chicago, Illinois

Fortunately, there's an easy way to create customer and file copies of a single invoice. You base your invoice report on a *cross-product query*, which is a special type of query that generates a separate record for each combination of two tables' records. Your query will cross the Orders table with a one-field table that stores the labels you want to print for each duplicate. The resulting query will create a

duplicate Orders record for each label. You can then base your invoice report on this query and include a text box to print the Whose Copy field. The report will print both customer and file copies for each invoice because the query supplies a record for each copy.

You create a cross-product query by including the two tables in the query—*without* joining them. Figure A shows the query Mr. Ritchie needs. This query selects several fields from the Orders table and then the single field from the Copies table. The query will repeat each Orders record for each record in the Copies table. ❖

Figure A



This query creates separate records for the customer and file copies of your invoices.

Inside MICROSOFT ACCESS 1993 Index

This comprehensive annual index is arranged by subject, listing the variety of tips and techniques covered in *Inside Microsoft Access* during 1993. The Cobb Group provides this index to make the collection of issues you received in 1993 more useful as a problem-

solving resource and to help you locate articles of interest in issues missing from your collection. You can order back issues at the price listed in the masthead by calling Customer Relations at (800) 223-8720.

Access Basic

Expressing a date as its day of the week, May 93
Creating the Proper() function to convert text to proper case, Jun 93
Parsing an Address field to derive Street Number and Street Name fields, Aug 93
A fast way to delete Import Errors tables, Oct 93
Understanding DLookup() and the other domain functions, Nov 93
Preventing users from adding new records while editing data, Dec 93
Saving money on postage by printing Postnet bar codes on envelopes, Oct 93
Understanding Access Basic's MsgBox() function, Aug 93
The NWIND database contains several useful utilities, Feb 93

Access keys

Defining access keys for controls on your forms, Jul 93
What happens to the ampersand in report labels?, Jun 93

Access upgrade information

Microsoft includes runtime version in its Access Distribution Kit, Sep 93
What enhancements does Access Version 1.1 bring?, Aug 93

append queries

Letting Access generate counter entries during an append query, May 93
Initialize counter fields with a simple append query, May 93

AutoExec macro

Disabling a database's AutoExec macro, Apr 93

Center Text button

Hints for centering labels on forms and in reports, Jun 93

clipboard

Loading Access data into a Word for Windows data file, Jul 93
Resetting a counter field to 1, May 93
Cutting, copying, and pasting in dialog boxes, May 93

combo boxes

Allowing blank values in combo boxes that limit entries to the list, Nov 93
Using a query as the row source of a combo box control, Sep 93
Combo box controls let you display a field other than the bound field, Sep 93

command buttons

This drop-and-drag shortcut lets you quickly create command buttons for macros, Feb 93
Creating command buttons to apply filters, Feb 93

Condition column in macros

Disabling individual macro statements, Jul 93

conditions in macros

Creating message boxes that offer choices during macros, Aug 93

count summary operator

Selecting records from one table that aren't in another table, Jul 93
Selecting records based on summary information, Jul 93

counter fields

Letting Access generate counter entries during an append query, May 93
Initialize counter fields with a simple append query, May 93
Resetting a counter field to 1, May 93
Can you use the Format property to set a counter field's starting value?, Jul 93
Five points to help you make the most of counter fields, Feb 93

crosstab queries

Understanding crosstab queries, Nov 93

database design

Navigational versus set-oriented databases, Apr 93
Five points to help you make the most of counter fields, Feb 93

database window

Don't give a query the same name as a table—or vice versa, Sep 93
Open Access objects in Design view by using keystroke shortcuts, Feb 93

Date/Time data type

In a report, you don't need DatePart() to group by date components, Nov 93
Using the DatePart() function in queries to group on date fields, Nov 93
Expressing a date as its day of the week, May 93

DatePart() function

Using the DatePart() function in queries to group on date fields, Nov 93

DCount() function

Checking for key violations as soon as you leave the primary key field, Dec 93

design tips

Sizing a label control to exactly the size of its contents, Jul 93
Defining access keys for controls on your forms, Jul 93
Combining text with field entries in form and report controls, Oct 93
Embossed text gives your forms a classic appearance, Oct 93
Problems with 3-of-9 bar codes, Dec 93
Laying out forms and reports quickly by using default controls, Dec 93
Saving money on postage by printing Postnet bar codes on envelopes, Oct 93
Hints for centering labels on forms and in reports, Jun 93
Creating command buttons to apply filters, Feb 93
Spruce up your forms with wallpaper, Apr 93
Protecting a table's key field while editing a datasheet, May 93
Setting a form's grid increments simplifies aligning and stacking controls, Jun 93
An undocumented way to select the form, May 93
Moving "fields" that comprise several controls, Apr 93
This drop-and-drag shortcut lets you quickly create command buttons for macros, Feb 93
Using parameters in reports to provide information as you print, May 93
Cancelling layout changes during form and report design, Feb 93

Microsoft Access Technical Support (206) 635-7050

Please include account number from label with any correspondence.

- DLookup() function**
Allowing blank values in combo boxes that limit entries to the list, Nov 93
- domain functions**
Understanding DLookup() and the other domain functions, Nov 93
- editing tips**
Preventing users from adding new records while editing data, Dec 93
Copying the previous record's entry saves time during data entry, Dec 93
- enable property**
Protecting a table's key field while editing a datasheet, May 93
- error handling**
Preventing users from adding new records while editing data, Dec 93
- Export... command**
A new export option in Access 1.1 lets you create Word for Windows data files, Aug 93
- expressions**
Combining text with field entries in form and report controls, Oct 93
- filters**
Using filters to change the sort order of forms, Feb 93
- fonts**
Using Windows 3.1's Character Map to paste special characters on your forms, Sep 93
Creating bar code labels in Access reports, Sep 93
- Format property**
Can you use the Format property to set a counter field's starting value?, Jul 93
- Format() function**
Expressing a date as its day of the week, May 93
- forms**
Using Query By Form to replace a parameter query's dialog boxes, Jun 93
Checking for key violations as soon as you leave the primary key field, Dec 93
Using the Query Parameters dialog box to manage parameters, May 93
Allowing blank values in combo boxes that limit entries to the list, Nov 93
Using a query as the row source of a combo box control, Sep 93
Combo box controls let you display a field other than the bound field, Sep 93
Spruce up your forms with wallpaper, Apr 93
Using filters to change the sort order of forms, Feb 93
- Freeze Columns command**
Protecting a table's key field while editing a datasheet, May 93
Freezing columns in a datasheet keeps important data in view, Apr 93
- GridX/GridY properties**
Setting a form's grid increments simplifies aligning and stacking controls, Jun 93
- Group By summary operator**
Selecting records based on summary information, Jul 93
- Help**
Context-sensitive Help is just a mouse click away, Dec 93
- identifiers**
Creating identifiers that reference controls on subforms, Jul 93
- Ilf() function**
Encoding ZIP+4 and Delivery Point codes in the Postnet system, Oct 93
- Import/Export**
Documenting your Import/Export specifications, Dec 93
- Importing macros lets you re-use other databases' utilities, Feb 93**
- joining tables in queries**
Understanding the outer join, Jul 93
- key violations**
Checking for key violations as soon as you leave the primary key field, Dec 93
- keystroke shortcuts**
Understanding Access' two-stage Undo feature, Feb 93
A fast way to delete Import Errors tables, Oct 93
Using Access' overstrike mode while editing cell and control entries, Nov 93
Context-sensitive Help is just a mouse click away, Dec 93
The [F2] key toggles the entry highlight in cells and controls, Aug 93
Disabling a database's AutoExec macro, Apr 93
Open Access objects in Design view by using keystroke shortcuts, Feb 93
- label controls**
Sizing a label control to exactly the size of its contents, Jul 93
- macros**
Disabling a database's AutoExec macro, Apr 93
Creating message boxes that offer choices during macros, Aug 93
Managing the billing and shipping addresses on an order form, Oct 93
The Confirm Delete macro lets you guard against accidental record deletions, Aug 93
The NWIND database contains several useful utilities, Feb 93
Printing a report for a single record, Sep 93
Importing macros lets you re-use other databases' utilities, Feb 93
This drop-and-drag shortcut lets you quickly create command buttons for macros, Feb 93
Disabling individual macro statements, Jul 93
Creating command buttons to apply filters, Feb 93
- mail merge**
A new export option in Access 1.1 lets you create Word for Windows data files, Aug 93
Loading Access data into a Word for Windows data file, Jul 93
- mailing label wizard**
Creating bar code labels in Access reports, Sep 93
Using the Mailing Label wizard, May 93
- mod operator**
Creating delivery-route reports by sorting even and odd street numbers, Aug 93
- MsgBox() function**
Understanding Access Basic's MsgBox() function, Aug 93
The Confirm Delete macro lets you guard against accidental record deletions, Aug 93
Creating message boxes that offer choices during macros, Aug 93
- OLE**
Spruce up your forms with wallpaper, Apr 93
- OpenForm action**
Managing the billing and shipping addresses on an order form, Oct 93
- Options... command**
Moving "fields" that comprise several controls, Apr 93
- outer join queries**
Understanding the outer join, Jul 93
Selecting records from one table that aren't in another table, Jul 93
- palette**
Embossed text gives your forms a classic appearance, Oct 93
- parameter queries**
Using Query By Form to replace a parameter query's dialog boxes, Jun 93
Using the Query Parameters dialog box to manage parameters, May 93
- Letting Access prompt you for query criteria, Apr 93**
- Print Setup... command**
Why Access sometimes prints a blank page between your report's pages, Aug 93
Creating a report with multiple "snaking" columns, Feb 93
- queries**
Letting Access generate counter entries during an append query, May 93
Using the Query Parameters dialog box to manage parameters, May 93
Selecting records based on summary information, Jul 93
Selecting records from one table that aren't in another table, Jul 93
Generating the correct title for your mailing labels, Nov 93
Encoding ZIP+4 and Delivery Point codes in the Postnet system, Oct 93
Understanding crosstab queries, Nov 93
Using the DatePart() function in queries to group on date fields, Nov 93
Don't give a query the same name as a table—or vice versa, Sep 93
Using a query as the row source of a combo box control, Sep 93
Parsing an Address field to derive Street Number and Street Name fields, Aug 93
Understanding the outer join, Jul 93
Initialize counter fields with a simple append query, May 93
Letting Access prompt you for query criteria, Apr 93
Using Query By Form to replace a parameter query's dialog boxes, Jun 93
A shortcut for including all of a table's fields in a query, Apr 93
Identifying table relationships on the QBE grid, Feb 93
- query fields**
Generating the correct title for your mailing labels, Nov 93
- report design**
Cancelling layout changes during form and report design, Feb 93
- report headers/footers**
Understanding the band-oriented nature of Access' report designer, Jun 93
- reports**
Printing a report for a single record, Sep 93
Creating a report that sorts and groups a name list as the phone book does, Apr 93
Understanding the band-oriented nature of Access' report designer, Jun 93
Creating delivery-route reports by sorting even and odd street numbers, Aug 93
Why Access sometimes prints a blank page between your report's pages, Aug 93
Using parameters in reports to provide information as you print, May 93
Understanding the Sorting and Grouping feature, Apr 93
Using the Mailing Label wizard, May 93
Creating bar code labels in Access reports, Sep 93
In a report, you don't need DatePart() to group by date components, Nov 93
Creating a report with multiple "snaking" columns, Feb 93
- SetVal action**
Managing the billing and shipping addresses on an order form, Oct 93
- Size To Fit command**
Sizing a label control to exactly the size of its contents, Jul 93
- Snap To Grid command**
Setting a form's grid increments simplifies aligning and stacking controls, Jun 93
- Sorting and Grouping**
Understanding the band-oriented nature of Access' report designer, Jun 93
In a report, you don't need DatePart() to group by date components, Nov 93
- Understanding the Sorting and Grouping feature, Apr 93**
Creating delivery-route reports by sorting even and odd street numbers, Aug 93
Creating a report that sorts and groups a name list as the phone book does, Apr 93
Creating a report with multiple "snaking" columns, Feb 93
- string manipulation**
Encoding ZIP+4 and Delivery Point codes in the Postnet system, Oct 93
Parsing an Address field to derive Street Number and Street Name fields, Aug 93
A problem with parsing an Address field to derive Street Number and Street Name fields, Dec 93
Creating the Proper() function to convert text to proper case, Jun 93
- subforms**
Creating identifiers that reference controls on subforms, Jul 93
- summarizing data**
Selecting records based on summary information, Jul 93
- support services**
Microsoft includes runtime version in its Access Distribution Kit, Sep 93
Downloading files from CompuServe, Sep 93
Get to know the Microsoft Knowledge Base, Jun 93
- system objects**
Documenting your Import/Export specifications, Dec 93
- table relationships**
Identifying table relationships on the QBE grid, Feb 93
- tables**
Don't give a query the same name as a table—or vice versa, Sep 93
Freezing columns in a datasheet keeps important data in view, Apr 93
Saving a table's layout, Apr 93
Five points to help you make the most of counter fields, Feb 93
- text box control**
Using parameters in reports to provide information as you print, May 93
- Undo**
Understanding Access' two-stage Undo feature, Feb 93
Cancelling layout changes during form and report design, Feb 93
- Val() function**
A problem with parsing an Address field to derive Street Number and Street Name fields, Dec 93
- Windows**
Ten tips for configuring Access and Windows for low-memory situations, Jun 93
Saving money on postage by printing Postnet bar codes on envelopes, Oct 93
Using SHARE to prevent simultaneous access to your database files, Nov 93
Problems with 3-of-9 bar codes, Dec 93
Using Windows 3.1's Character Map to paste special characters on your forms, Sep 93
Are your wizards missing?, May 93
- Windows for Workgroups**
Using SHARE to prevent simultaneous access to your database files, Nov 93
- wizards**
Using the Mailing Label wizard, May 93
Are your wizards missing?, May 93
- Word for Windows**
A new export option in Access 1.1 lets you create Word for Windows data files, Aug 93
Loading Access data into a Word for Windows data file, Jul 93

